

Lab 3. Loops

Homework

The homework for this week consists of 5 multiple choice questions organised as a test in eLearning. You need to answer them online, via eLearning. Login to eLearning, go to **Course Material** (left panel) and click on the link **w3-homework** to start the online test (homework). The same rules and hints as last week:

1. Do not delay the submission of the homework till the very last moment as eLearning may get overloaded and you may miss the deadline! You will not be able to submit after the deadline.
2. If you have problems submitting in eLearning before the deadline, email your answers to sit.info1103@sydney.edu.au (e.g. 1A, 2B, etc.).
3. You are allowed 2 attempts. If you submit twice, the second attempt will be marked.
4. A note on incomplete submissions: Incomplete submissions are not valid submissions and are ignored. An incomplete submission is a submission started before the deadline but not submitted before the deadline, e.g. you started before the deadline but pressed “Submit” after the deadline or you started and finished before the deadline but forgot to press “Submit”.

Only complete (valid) submissions are marked. For example, if you have attempted the homework twice and your first submission was complete but your second submission was incomplete, then there is only one valid submission (the first one) and it will be marked.

Hence, start working on the homework early so that you can finish it before the deadline, and remember to press “Submit”!

Here are the questions for your information (remember that they must be answered online via eLearning!):

1. How many times will the following loop run?

```
int i = 0;
while (i < 10)
{
    System.out.println(i);
    i++;
}
```

- A) 0
- B) 8
- C) 9
- D) 10

2. What will this loop print?

```
int count = 0;
do
{
    System.out.println(count);
    count++;
}
```

```

} while (count < 0);
System.out.println("count after loop = " + count);

```

- A)
0
count after loop = 1
- B)
0
count after loop = 0
- C)
1
count after loop = 1
- D) count after loop = 0

3. What will this loop print?

```

for (int n =1; n <=4; n++)
    System.out.print(n);

```

- A) 123
- B) 1234
- C) 12345...(infinite loop)
- D) 1

4) Which loop does not check a condition at the beginning of the loop?

- I. The `do-while` loop
- II. The `while` loop
- III. The `for` loop

- A) I and II
- B) I and III
- C) I only
- D) III only

5) Which of the following loops may not enter the loop body at all?

- I. `for` loop
- II. `while` loop
- III. `do-while` loop

- A) I only
- B) I and II only
- C) II and III only
- D) I and III only

1. Submission using PASTA

Task 1, Task 2, and the assignment need to be submitted via the online system PASTA. We will write a simple program and submit it via PASTA to make sure that everyone knows how to do this.

1. Go to PASTA and login with your unikey login and password:
<http://info1103.it.usyd.edu.au/>
2. Click on **Simple Calculator** and read the description of the task:

Week 3

[Simple Calculator](#) - 0.0 / 0.0
Due: Mon 30 November 2015 at 00:00
 ∞ submissions allowed
No attempts on record.

Assessment Description

Write a Java program to read in an integer number (as a String) from standard input, and print the value of the following calculation, where n is the number:

- if $n < 0$ - print out $n^2 + n/2$
- if $n \geq 0$ and is **even** - print out $n^3(n-1)$
- if $n \geq 0$ and is **odd** - print out $n^{-1} + 3n$

Sample Output

Input:

java SimpleCalc 10

Output:

9000

Input:

java SimpleCalc 11

Output:

33.090909

Input:

java SimpleCalc -100

Output:

9950

3. Let's write a solution for this exercise and submit it in PASTA.

In Eclipse, create a package "lab3" in your "Labs" project, and then a class "SimpleCalc.java" within the package.

Download the file `SimpleCalc.java` from eLearning/lab notes and paste it in your class with the same name. The code is also provided below. Although the code compiles and run, it is not completely correct - there is a mistake in case 3:

```

package lab3;

/**
 * This program reads an integer from the keyboard
 * and computes:
 * 1)  $n^2 + n / 2$  if  $n < 0$ 
 * 2)  $n^3 * (n - 1)$  if  $n \geq 0$  and  $n$  is even
 * 3)  $1 / n + 3 * n$  if  $n \geq 0$  and  $n$  is odd
 */

import java.util.Scanner;

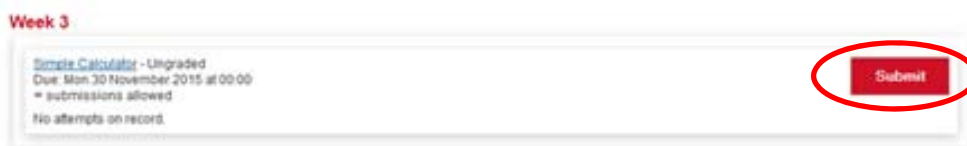
public class SimpleCalc
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter an integer number: ");
        int n = in.nextInt();

        if (n < 0)
        {
            System.out.println(n * n + n / 2.0);
        }
        else if (n % 2 == 0)
        {
            System.out.println(n * n * n * (n - 1));
        }
        else
        {
            //System.out.println(1.0 / n + 3 * n); //correct version
            System.out.println(n); //incorrect
        }
    }
}

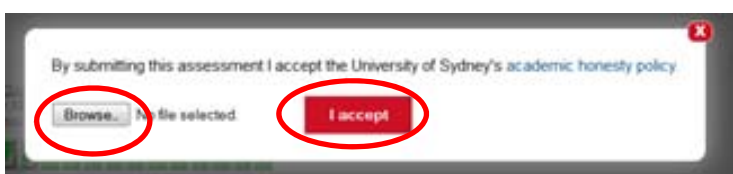
```

Now let's submit the SimpleCalc.java file to PASTA:

1. Click on "Submit":



2. Click "Choose File", locate your SimpleCalc.java, and then press "I accept" after reading the University's academic honesty policy:



3. You should see a message indicating that your code is queued for testing. In about a minute your code will be tested by our testing program. It may take longer if there are many students submitting at the same time.

If you see a red error box, you need to fix any problems indicated before your submission will be accepted.

Once your program has been tested, your screen will change and you will see green, red and grey boxes below this message. Each box corresponds to a test; a red box indicates that your program has failed the respective test, a green box indicates that your program has passed the test, and a grey box indicates that the test is hidden (you will not see the result of the test until after the due date). For this exercise there are no hidden tests.

How many tests did you pass? How many tests did you fail? If you have red boxes (you should), you can see which tests were not passed by clicking on the name of the task (“Simple Calculator” in this case).

Now let’s fix the errors. Go to Eclipse, modify the code so that it prints the correct output for case 3 and submit again. Did you pass all the tests? If so, you will get the full mark! Well, there are no marks for this exercise, but there are for Task 1.

So now you know how to submit your tasks in PASTA!

Note: Remember that to access PASTA from home (off-campus) you need to install a program called “VPN client” and you also need to run it every time when you try to access PASTA. For more information how to do this, please see the INFO1103 User Manual.

2. A reminder

For the programming exercises below, remember to create a package “lab3” in the Java project “Labs”. Then create a class within this project and package for each exercise below.

3. Sum using different types of loops

There is a famous story about a primary school teacher who wanted to occupy his students’ time by making the children compute the sum of $1 + 2 + 3 + \dots + 100$ by hand. As the story goes, the teacher was astounded when one of the children immediately produced the correct answer: 5050. This student was Carl Gauss, who grew up to be one of the most famous mathematicians of the eighteenth century.

Repeat Gauss’s calculation by writing a loop that will compute and print the above sum. Write three different versions of this program: using a for loop, while loop and a do-while loop. Name your programs `SumFor.java`, `SumWhile.java` and `SumDoWhile.java`. Which type of loop is most appropriate for this task?

4. Nested for loops

Write a program called `NestedLoops.java` that prints the following using nested **for** loops:

```
1 1
1 2
1 3
1 4
```

```
2 1
2 2
2 3
2 4
```

5. A simple while loop with sentinel

Write a program called `NonNegativeSumSentinel.java` that uses a `while` loop with a sentinel value. It computes the sum of a list of non-negative values, that are entered by the user. For example, these values might be the number of hours worked by each person in a team, and we know that they can't be negative. The program prompts the user to enter a sequence of whole nonnegative numbers and a negative number to quit. Thus, it uses the negative number as a sentinel value to signal the end of the list. Recall that the sentinel value should be different than the expected values; since we expect non-negative values, a negative sentinel is appropriate.

When the user enters a negative value, the program stops and computes the sum of the numbers. Here is a sample output:

```
Enter nonnegative whole numbers and a negative number at the end to quit:
2 3 4 5 -1
The sum of the numbers is 14
```

6. A simple while loop with sentinel and Boolean variable

Modify the previous program so that it uses a Boolean variable to end the loop. Your tutor will guide you. Save your program as `NonNegativeSumSentinelBoolean.java`

7. Another while loop with sentinel – to be done at your own time

Write a program called `SumWithSentinel.java` that uses a `while` loop with a sentinel value. In each iteration, it prompts the user to enter a number (read this number as `double`) and keeps the sum of the numbers so far and their count. The program stops whenever the user enters “q” to quit. When this happens, the program prints the sum and number of entries. Here is a sample output:

```
Enter a number or type q to quit:
40
Enter a number or type q to quit:
6
Enter a number or type q to quit:
q
The total is 56.0.
You entered 3 values.
```

Hint: Use the method `hasNextDouble` from the class `Scanner` to read the entry. It returns true if the entry is a number and false otherwise (i.e. if the entry is “q”).

8. Number of vowels

Write a program called `NumVowels.java` that reads a word as string and prints the number of vowels in it. There are 5 vowels in English: a, e, i, o and u. Assume that the word contains only lower-case characters. Here is a sample output:

```
Enter a word with no spaces: abracadabra
Number of vowels: 5
```

```
Enter a word with no spaces: quickbrownfox
Number of vowels: 4
```

9. Challenge

Suppose that there is a party. When a new person arrives, he or she shakes hands with everyone who is already there. Write a program called `Handshakes.java` that asks the user how many people attended the party and then calculates the total number of handshakes that occurred for this number of people. Here is a sample output:

```
How many people attended the party?
15
The total number of handshakes is 105.
```

Hint: The key question is: If k people attend the party, how many handshakes will occur?